Goddard
Space S
Flight P
Center I

NASA
GSFC

Managing Requirements (Without Those Expensive Tools)

Presented by: Ella Page

Software Process Improvement (SPI) Project

Purpose and Objectives

- **Purpose: Provide guidance on managing requirements per NPR 7150.2**

- **Objective - After this class you should understand:**
  - **What the planning considerations are for requirements management**
  - **Approaches to defining requirements and getting them approved**
  - **How to manage requirements change**
  - **What requirements bi-directional traceability is and how to maintain it**
  - **What requirements management records should be maintained**

SWE-049   The *project* shall document the software requirements.

SWE-050   The *project* shall identify, develop, document, approve, and maintain software requirements based on analysis of customer and other stakeholder requirements and the operational concepts.

SWE-051   The *project* shall perform software requirements analysis based on flowed-down and derived requirements from the top-level systems engineering requirements and the hardware specifications and design.

SWE-052   The *project* shall perform, document, and maintain bi-directional traceability between the software requirement and the higher level requirement.

SWE-053   The *project* shall collect and manage changes to the software requirements. Note: The project should analyze and document changes to requirements for cost, technical, and schedule impacts.

SWE-054   The *project* shall identify inconsistencies between requirements and project plans and software products and initiate corrective actions.

SWE-055   The *project* shall perform requirements validation to ensure that the software will perform as intended in the customer environment. Note: This should include confirmation that the requirements meet the needs and expectations of the customer.

- **How will your requirements be managed?**
  - **How will you define and document them?**
  - **How will you validate them?**
  - **Where will you maintain the requirements repository?**
  - **How will you control requirement change?**
  - **What kind of tools will you need?**
- **Who will be responsible for these activities?**

Defining the Requirements

- **Understand your source requirements**
  - **Know who provides them or where they come from**
  - **Analyze source requirements to make sure they are clear, concise, and verifiable**
  - **Develop an understanding of what the requirements really mean**
  - **Factor in customer and other stakeholder requirements**
- **Derive detailed requirements through analysis as needed**
  - **Start with source requirements, hardware requirements, and operational concepts**
- **Document detailed requirements and place them in a repository**

- **Conduct peer reviews to validate the result**
  - **Peer reviews are required for requirements by NPR 7150.2**

> See *http://software.gsfc.nasa.gov/tools.cfm* and search on "inspection" to find two useful tools:
> Inspection Moderator's Tool
> Inspection Metrics Tool

Types of Peer Reviews

	Inspections	Walkthroughs	Reading
Why you do the review	*Defect detection*	*Communication Defect detection*	*Defect detection*
Advance review of material	*Required*	*Optional*	*Optional*
Use of checklists or reading techniques	*Included*	*Optional*	*Optional*
Meeting leader	*Moderator*	*Author*	*No meeting held*
Manager participation	*NOT ALLOWED!*	*Optional*	*Optional*

Assets to help you:
Requirements Peer Review Checklist
         (*http://software.gsfc.nasa.gov/AssetsApproved/PA2.2.1.5.doc*)

Sub-process *Inspections, Peer Reviews, and Walkthroughs*
         (*http://software.gsfc.nasa.gov/AssetsApproved/PA2.5.0.doc*)

- **Ways to obtain approval of the requirements**
  - **Use formal reviews, like a System Requirements Review (SRR)**
  - **Use reviews of the requirements document**
- **Document acceptance and approval (both team and customer)**
  - **Get written approvals … and keep them in your repository**
  - **Obtain signatures on requirements documents**

Repository Tools

- **Use a tool to store and manage your requirements**
- **Commercial Off-the-Shelf (COTS) tools are available, but are usually expensive**
  - **RequisitePro**
  - **Telelogic DOORS®**
  - **Slate**
- **SPI provides a free tool**
  - **It is simpler than the COTS tools, but free**

Make sure your tool supports bi-directional traceability

Example of a Simple Requirements Repository

Requirement Identifier	Requirement text	Requirement Source	Requirement State
F3308	The ABC software shall initialize itself following a Cold Restart of the main processor.	MRD 3.22	Have all info
F3309	On cold restart, the software shall initialize all ABC data segments, causing all variables to be zeroed or reset to their preset values stored in EEPROM.	CR 15	Req deleted
F3310	The software shall reset all ABC database-specifiable parameters to their preset values stored in EEPROM.	MRD 3.23	Have all info
F3311	The software shall set all ABC telemetry and command output packet buffers to zero.	MRD 3.24	Have all info
F3312	On cold restart, the ABC software shall enter Sun Acquisition Mode.	MRD 3.25	Have all info
F3313	On cold restart, the ABC software shall initialize ABC unique parameters as appropriate to ensure that each ABC software process starts from a known state.	CR 23	CR Submitted
F3314	The ABC software shall set all ABC statistics and the statistics reset counter to zero.	MRD 3.26	Have all info
F3323	The ABC software shall initialize prior solutions for all sensor and actuator data processing results.	MRD 3.27	Have all info
F3324	The ABC software shall set the "CSS in eclipse" flag.	CR 67	Have all info
F3325	The ABC software shall flag the prior GT-implied spacecraft pointing error solutions as invalid.	MRD 3.28	Have all info

Find the SPI Requirements Traceability Tool at:

software.gsfc.nasa.gov/tools.cfm

Implementing Requirements

- **Allocate requirements to project components**
  - *Computer Software Configuration Items* (CSCIs)
  - **Software modules or units**
- **Allocate requirements to component builds**
  - **Include the "framework" in early builds**
  - **Consider relative size of builds**
- **Ensure assigned personnel understand the requirements**
- **Document component allocation information (in the repository or a traceability tool)**
- **Validate the implementation of each requirement through testing or inspection**
  - **Identify the tests for each requirement in the traceability matrix**

Expanding the Repository

Sort Field	Requirement Identifier	Requirement text	Requirement Source	Requirement State	CSCI	CSCI Build Number	Module or Unit
1	F3308	The ABC software shall initialize itself following a Cold Restart of the main processor.	MRD 3.22	Have all info	ABC	1	Unit 1
2	F3309	On cold restart, the software shall initialize all ABC data segments, causing all variables to be zeroed or reset to their preset values stored in EEPROM.	CR 15	Req deleted			
3	F3310	The software shall reset all ABC database-specifiable parameters to their preset values stored in EEPROM.	MRD 3.23	Have all info	ABC	1	Unit 1
4	F3311	The software shall set all ABC telemetry and command output packet buffers to zero.	MRD 3.24	Have all info	ABC	1	Unit 1
5	F3312	On cold restart, the ABC software shall enter Sun Acquisition Mode.	MRD 3.25	Have all info	XYZ	1	Unit 12
6	F3313	On cold restart, the ABC software shall initialize ABC unique parameters as appropriate to ensure that each ABC software process starts from a known state.	CR 23	CR Submitted	ABC	2	Unit 2
7	F3314	The ABC software shall set all ABC statistics and the statistics reset counter to zero.	MRD 3.26	Have all info	ABC	2	Unit 2
8	F3323	The ABC software shall initialize prior solutions for all sensor and actuator data processing results.	MRD 3.27	Have all info	XYZ	1	Unit 13
9	F3324	The ABC software shall set the "CSS in eclipse" flag.	CR 67	Have all info	XYZ	1	Unit 17
10	F3325	The ABC software shall flag the prior GT-implied spacecraft pointing error solutions as invalid.	MRD 3.28	Have all info	ABC	1	Unit 2

Managing Requirements Change

- **Managing requirements change is linked closely to the Configuration Management (CM) process**
  - **Document change requests using a change request form***
  - **Track handling of the requests using a change request log***
  - **Analyze requests for feasibility, cost and schedule impacts, and other criteria as appropriate**
  - **Use your Configuration Control Board (CCB) as the approving body**
  - **Obtain approval at higher levels as appropriate**
  - **Add new or changed requirements to the repository**
  - **Assign accepted changes to the appropriate CSCI, build, and units**

Find change request tools at:
http://software.gsfc.nasa.gov/tools.cfm

A Change Request Form

Change Request Form

Requestor fills out this section

Project:

System(s)/subsystem(s):

Requestor:
Name: Date initiated:

Urgency:
☐ Routine
☐ Urgent

Item type:
☐ Requirement
☐ Document
☐ Process

Current version of item:
Item number (if appropriate):
Description of existing item (enter "none" if request is for a new item):

New version of Item:
Description of new version of item (enter "delete" to delete an item):

Suggested new item number (optional):

Rationale:

Request Disposition:
Tracking Number: Date Submitted:

Analysis Assigned to: Date Assigned:

Analysis:

Impact:

Feasibility:

Disposition date:

Result:
☐ Accepted
☐ Accepted with modification

If accepted with modification, describe the modification:

☐ Rejected

If rejected, rationale for rejection:

Signature:
Approved or rejected by: Date:

Work Orders/Action Items Assigned To Perform Change:

WO or AI Number	Assignee	Date Assigned

A Change Request Log

1	Change Request Log						
2	Project Name:						
3							
4	Tracking Number	Item Type	Systems/Subsystems	Short Description of Change	Urgency	Requestor	Date Submitted
5							
6							
7							

Analysis Assigned To:	Result	WO & AI #'s	Signature Date

Maintaining Bi-directional Traceability

- **Bi-directional Traceability allows you to**
  - Trace down from source requirement to lower-level (derived) requirements and into the design, implementation, and test
  - Trace back up from the design, implementation, and test, to derived and source requirements
- **Bi-directional traceability helps determine that**
  - All source requirements have been addressed
  - All lower-level requirements can be traced to a valid source
- **Requirements traceability helps identify relationships to other entities, such as**
  - Intermediate and final work products
  - Changes in design, implementation, or test plans

Maintaining Bi-directional Traceability Throughout the Life Cycle

As part of requirements development …

… as part of product design, implementation, and maintenance …

… and as part of product verification

Sort Field	Requirement Identifier	Requirement text	Requirement Source	Requirement State	CSCI	CSCI Build Number	Module or Unit	Build Test Procedure	System Test Procedure
1	F3308	The ABC software shall initialize itself following a Cold Restart of the main processor.	MRD 3.22	Have all info	ABC	1	Unit 1	ABC B1-8	ABC ST-2
2	F3309	On cold restart, the software shall initialize all ABC data segments, causing all variables to be zeroed or reset to their preset values stored in EEPROM.	CR 15	Req deleted					
3	F3310	The software shall reset all ABC database-specifiable parameters to their preset values stored in EEPROM.	MRD 3.23	Have all info	ABC	1	Unit 1	ABC B1-2	ABC ST-1
4	F3311	The software shall set all ABC telemetry and command output packet buffers to zero.	MRD 3.24	Have all info	ABC	1	Unit 1	ABC B1-3	ABC ST-3
5	F3312	On cold restart, the ABC software shall enter Sun Acquisition Mode.	MRD 3.25	Have all info	XYZ	1	Unit 12	XYZ B1-4	XYZ ST-2
6	F3313	On cold restart, the ABC software shall initialize ABC unique parameters as appropriate to ensure that each ABC software process starts from a known state.	CR 23	CR Submitted	ABC	2	Unit 2	ABC B2-1	ABC ST-2
7	F3314	The ABC software shall set all ABC statistics and the statistics reset counter to zero.	MRD 3.26	Have all info	ABC	2	Unit 2	ABC B2-2	ABC ST-1
8	F3323	The ABC software shall initialize prior solutions for all sensor and actuator data processing results.	MRD 3.27	Have all info	XYZ	1	Unit 13	XYZ B1-1	XYZ ST-4
9	F3324	The ABC software shall set the "CSS in eclipse" flag.	CR 67	Have all info	XYZ	1	Unit 17	XYZ B1-2	XYZ ST-3
10	F3325	The ABC software shall flag the prior GT-implied spacecraft pointing error solutions as invalid.	MRD 3.28	Have all info	ABC	1	Unit 2	ABC B1-4	ABC ST-4

- **Bi-directional traceability requires that you be able to show the trace in both directions**
  - **Requirements down through design and test**
  - **Test and design up to requirements**
- **To do this in the "free" SPI tool you must have one CSCI, unit, or test identifier in a cell**
  - **Each requirement should map to one CSCI, one unit, and one test (per build)**
  - **If it maps to more consider breaking the requirement down further**

> *Or … put the requirement in twice showing the multiple mappings*

Using the Requirements Traceability Tool: http://software.gsfc.nasa.gov/tools.cfm

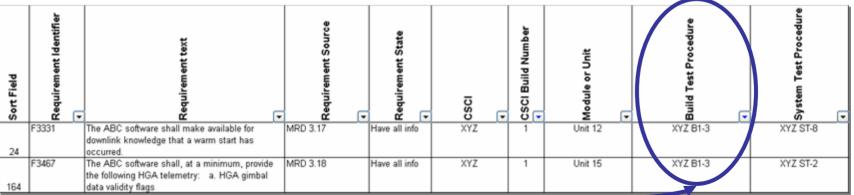| Sort Field	Requirement Identifier	Requirement text	Requirement Source	Requirement State	CSCI	CSCI Build Number	Module or Unit	Build Test Procedure	System Test Procedure
125	F3428	For the controlling GT only, if the GT data set has been flagged as valid (acquisition), the ABC software shall convert the GT-implied sun direction vector to a GTimplied spacecraft pointing (Y, Z) error.	MRD 3.05	Req is TBD	ABC	2	TBD		
126	F3429	For the controlling GT only, if the GT data set has been flagged as valid (acquisition), if the GT data for the preceding cycle has been flagged as valid (acquisition), and if the controlling GT designation has not changed between cycles, the ABC software shall compute the GTimplied body rate.	MRD 3.06	Req is TBD	ABC	3	TBD		

Filter on columns to get specific information … like TBD requirements

Sort Field	Requirement Identifier	Requirement text	Requirement Source	Requirement State	CSCI	CSCI Build Number	Module or Unit	Build Test Procedure	System Test Procedure
24	F3331	The ABC software shall make available for downlink knowledge that a warm start has occurred.	MRD 3.17	Have all info	XYZ	1	Unit 12	XYZ B1-3	XYZ ST-8
164	F3467	The ABC software shall, at a minimum, provide the following HGA telemetry: a. HGA gimbal data validity flags	MRD 3.18	Have all info	XYZ	1	Unit 15	XYZ B1-3	XYZ ST-2

Get bi-directional traceability by filtering on CSCI, Build, Test Procedure, etc.

When You See a Problem With Requirements …

- **Review requirements (and changes) for consistency with other plans and constraints**
    - During development or change analysis
- **Look for inconsistencies between what requirements state and what is in place**
    - Current plans (e.g. cost, schedule, staffing)
    - Associated requirements (e.g., interface or performance)
    - Designs that have been baselined
    - Implementation that is done or in progress
    - Components that have been tested
- **Document any inconsistencies found**
- **Resolve any inconsistencies**

Requirements Measures

- **Per NPR 7150.2 you must collect requirement volatility measures**

- **Recommended measures are the following by both CSCI and system:**
  - **Total number of requirements**
  - **Number of new requirements**
  - **Number of changed requirements**
  - **Number of deleted requirements**

- **See the SPI Website for Requirements Measures tools**
  - **Go to *http://software.gsfc.nasa.gov/tools.cfm* and search on "Requirements Metrics Tool"**

Products of the requirements management process that should be kept in the project data stores:

- **Change Request Forms**
- **Change Request Log**
- **Requirements Documents**
- **Meeting Minutes**
  - **Meetings of understanding**
  - **CCB meetings**
  - **Requirements reviews and inspections**
- **Email messages related to requirements**
- **Requirement analysis documents**
- **Artifacts changed by inconsistencies (before and after versions)**

- **Use some kind of a tool or form to manage requirement change requests**
- **SPI tools are not required, but are there if you need them**
  - **Are compliant with CMMI**
  - **Are available at http://software.gsfc.nasa.gov/tools.cfm**

Summary

- **Plan how requirements will be managed**
- **Understand your source and derived requirements and get documented approval**
- **Identify where each requirement will be implemented and tested**
- **Manage requirement change requests**
- **Maintain bi-directional traceability of requirements**
- **Make sure you**
  - **Resolve inconsistencies with other plans**
  - **Use tools as needed**
  - **Collect required measures**
  - **Keep related artifacts**

Questions?

- **CCB – Configuration Control Board**
- **CM – Configuration Management**
- **CMMI – Capability Maturity Model Integrated**
- **COTS – Commercial Off-the-Shelf**
- **CSCI – Computer Software Configuration Item**
- **NPR – NASA Procedural Requirement**
- **SPI – Software Process Improvement**
- **SRR – System Requirements Review**